# Machine Learning for Supermarket Sales Forecast

**Rising Odegua Mike Omogbhemhe**

Department of Computer Science Ambrose Alli University

Ekpoma, Edo state, Nigeria

**Abstract**

Many supermarkets today do not have a strong forecast of their yearly sales. This is mostly due to the lack of the skills, resources and knowledge to make sales estimation. At best, most supermarket chain store use adhoc tools and processes to analyze and predict sales for the coming year. The use of traditional statistical method to forecast supermarket sales has met lot of challenges unaddressed and mostly result in the creation of predictive models that perform poorly.

The era of big data coupled with access to massive compute power has made machine learning a goto for sales forecast. In this paper, we investigate the forecasting of sales with new and powerful machine learning algorithms and compare it with traditional and simple linear regression models. Two different methods used are Gradient Boosting and Random forest. These two methods are compared side by side against a simple linear regression.

The data used to train the machine learning models is a data provided by Data Science Nigeria on the Zindi platform, the data is collected from a supermarket chain called “Chukwudi Supermarkets”.

The results show that the newer machine learning method perform far better than the simple linear regression. Also, the most important variables that helped in better forecast are “Supermarket\_Type\_Grocery\_Store”, “Product\_Price” and “Supermarket\_Opening\_Year”.

**1 Introduction**

The goal of every supermarket is to make profit. This is achieved when more goods are sold and the turnover is high. A major challenge to increasing sales of a supermarket lies in the ability of the manager to forecast sales pattern and know readily before hand when to order and replenish inventories as well as plan for manpower and staffs.

The amount of sales data has steadily be on the increase in recent years and the ability to leverage this gold of data separates high performing supermarket from the others. One of the most valuable asset a supermarket can have is data generated by customers as they interact with various supermarkets. Within these data, lies important patterns and variables that can be modeled using a machine learning algorithm; and this can to a very high degree of accuracy correctly forecast sales [1, 2].

There exist several techniques to forecasting supermarket sales and historically, many supermarkets have relied on these traditional statistical models [3]. However, machine learning has grown to be an important area of data science that has gained ground due to its high predictive and forecasting powers and as such as become the go-to for highly accurate sales forecasting as well as other important areas [3, 4, 5].

To correctly forecast a future event, a machine learning model is trained on data from which it learns patterns that are used to predict future instances. An accurate forecasting model can greatly increase supermarket revenue and is generally of great importance to the organization as it improves profit as well as provides insights into the way customers can be better served [3].

**1.1 Motivation**

The main aim of this research work is evaluate new machine learning techniques for sales forecasting to simple traditional methods. In doing so, we try to answer the following questions:

1. Which of Linear model, Gradient Boosting and Random Forest is the best in terms of:
   1. Lower mean absolute error
   2. Training time
2. What variables are the most important when forecasting sales?
3. What is the impact of supermarket type in forecasting sales?
4. What is the impact of product price in forecasting sales?

**2 Background**

**2.1 Multiple Linear Regression Model**

Multiple linear regression is an extension of a simple linear regression where there are *p* independent variables, as simple linear regression can be thought of as a special case of multiple linear regression, where *p*=1. The term ‘linear’ is used because in multiple linear regression, it is assumed that y the dependent variable is directly related to a linear combination of the independent variables. The relationship between the dependent variable and the independent variables is represented by the following equation:

*yi = β0 + β1x1i + β2x2i + … + βpxpi+ ei*

Where: *β0* is the constant term and *β1* to *βp* are the coefficients relating the *p* independent variables to y-variable of interest.

**2.2 Gradient Boosting Model**

Boosting is a popular machine learning algorithm that falls under the umbrella of ensembles. Boosting was introduced by (Kearns and Valiant, 1989), in answer to the question whether a “weak learner” could be made better by using some form of modification. This was discovered to be possible by Schapire (1990) and the first boosting algorithm Adaptive Boosting (AdaBoost) was created by (Freund and Schapire, 1996).

The concept of boosting is to correct the mistakes made by previous learners and improving on those areas Zhou (2012).

Boosting can also be seen as a kind of stage wise “additive modeling” (Buhlmann and Hothorn) in that it is an additive combination of a simple base estimator.

Gradient Boosting (Friedman 2000) is a type of boosting where the objective is treated as optimization problem and training is done using weight updates by gradient descent.

**Gradient Boosting Algorithm**

1. Specify the following as input:
   1. Input data N
   2. Number of iterations M
   3. A base-learner h
   4. A loss function l
2. Initialize l0 to a constant
3. for t = 1 to M:

compute the negative gradient

1. fit a new base-learner function hi
2. Find the best gradient descent step-size p
3. update the function estimate

**2.3 Random Forest Model**

Random forest is a tree-based machine learning algorithm introduced by (Breiman 2001). In random forest, multiple decision trees are constructed and trained on a bootstrap sample drawn from the original dataset. The final result in the case of regression task, is an average of the individual predictions from each decision tree, and a majority class vote in a classification task.

(Breiman 2001) defined Random Forest as a classifier consisting of a collection of trees structured classifiers {h(**x**,Θk ), k=1, ...} where the {Θk} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x.

Numerous empirical studies (Breiman, 2001; Svetnik et al., 2003; Diaz-Uriarte and de Andres, 2006), shows that random forests are serious competitors to state-of-the-art methods such as boosting (Freund and Shapire, 1996). Most industry practitioners consider it to be one of the most accurate general-purpose learning techniques. Random Forest are fast and easy to implement, produce highly accurate predictions and can handle a very large number of input features without the risk of overfitting.

**Random Forest Algorithm**

The random forest algorithm for both classification and regression task is shown below:

1. Draw n bootstrap samples from the original dataset.
2. For each ni bootstrap sample, grow a classification or regression tree, by choosing the best split among *m* randomly selected variables.
3. Predict new data by aggregating the predictions of the n trees using average for regression and majority voting for classification.

**3 Methodology**

The models compared in this study (Multiple Linear Regression, Gradient Boosting and Random Forest) have been used for numerous problems in different forecasting task and have been chosen based on their popularity in industry. In addition, the data used in this study is provided by Data Science Nigeria, a Data Science and Artificial Intelligence Hub as part of their machine learning competitions.

**3.1 The Data**

The data consist of numerous supermarket variables like opening year, product prices, supermarket location etc. The data set contains a sample of 4990 instances with 13 features/variables. The description of the data is shown in table 1.

|  |  |  |
| --- | --- | --- |
| Data Feature | Description | Feature Type |
| Product\_Identifier | Unique identifier for each product | String |
| Supermarket\_Identifier | Unique identifier for each supermarket | String |
| Product\_Supermarket\_Identifier | Combination of product and supermarket identifiers | String |
| Product\_Weight | The weight of a product | Numeric |
| Product\_Fat\_Content | The fat content contained in a product. | Categorical |
| Product\_Shelf\_Visibility | A numeric value that captures the visibilty of a product. | Numeric |
| Product\_Type | The type of product. | Categorical |
| Product\_Price | The selling price of a product. | Numeric |
| Supermarket\_Opening\_Year | The year the supermarket was opened. | Numeric |
| Supermarket\_Size | The size of a supermarket | Categorical |
| Supermarket\_Location\_Type | The location of a supermarket | Categorical |
| Supermarket\_Type | The type of the supermarket | Categorical |
| Product\_Supermarket\_Sales | The sales made by the supermarket (**Target Feature)** | Numeric |

Table 1

**3.2 Data Processing and Engineering**

After extensive data cleaning and processing, three features (Product\_Identifier, Supermarket\_Identifier, Product\_Supermarket\_Identifier) were removed as they are unique IDs and add little or no effect to our model’s performance. Further exploration of the dataset showed the need to create new features from the existing ones. This process is termed feature engineering; The new features created are:

1. is\_normal\_fat: Groups the feature *Product\_Fat\_Content* into two groups 0 and 1
2. open\_in\_the\_2000s: Groups the feature S*upermarket\_Opening\_Year* into two classes.
3. Product\_type\_cluster Clusters the *Product\_Type* into two classes

Next, we use one-hot-encoding scheme to encode all categorical variables, filled missing instances in Product\_Weight feature with the mean and finally standardize our data set by subtracting the mean and then dividing by the standard deviation.

The three models where trained on the data set and a 10 fold cross validation strategy was used since the data set was limited. The mean absolute error was recorded as performance metrics.

**3.3 Linear Regression Model**

The multiple linear regression used is implemented in sklearn, a Python machine learning library. The default parameters where used as the model required minimal parameter tuning and works well out-of-the-box.

**3.4 Gradient Boosting Model**

For gradient boosting, we set the number of boosted trees (n\_estimators) to 200, *max\_depths* to 6, *max\_features* as square root and the mini\_sample\_split to 4. All other parameters where left as default.

**3.5 Random Forest Model**

For random forest model, we specify the number of trees (n\_estimators) to 100 and the *max\_depth* to 5. All other parameters where left as default.

**3.6 Performance Metric**

We use the mean absolute error (MAE) in model evaluation. This means that a lower MAE results in a better model. The choice of performance metric is based on the fact that the task is a regression task and the MAE is a tested and trusted metric that gives a good measure of model performance.

**3.6.1 Mean Absolute Error**

Mean absolute error (MAE)is a measure of difference between two continuous variables. Assume *X* and *Y* are variables from an observations, say X is the known value and Y is the predicted value from a machine learning model, the Mean Absolute Error (MAE) is the average vertical distance between each observed point and the predicted point.

The mean absolute error is given by:

MAE =

**4 Results**

In this section the results of the three models will be presented. The results were obtained by applying the three models; LR, GB and RF on a 10-fold cross validation dataset.

**4.1 Error Measures & Standard Deviations**

In table 2, the MAE and standard deviations are shown respectively. In most of the runs the RMSEs of the MLP is considerably larger than the RMSE of the other models. This indicates that the MLP consistently fails to match large deviations in the actual sales. The MLP also shows a larger average MAPE than the other models and a high of the MAPE also informs us that the model provides deviating predictions. The same can be stated for the of the RMSE. Table 4.1 shows that the SVM provides the lowest average value of the MAPEs. This in conjunction with the low of the MAPE indicates that there is a lower deviation between the predicted and the acutal sales than the other models. Moreover the predictions by the SVM also has the lowest of the RMSE which further emphasizes a more consistent performance than the other models.

**5 Discussion**

**5.1 Discussion of results**

In order to verify our problem statement we chose to use the MAPE and RMSE as error measures. As the SVM scored a lower average MAPE and RMSE than the MLP and the RBFN, our result is similar to the one obtained by Pillo et al. [25], who showed that the SVM outperformed a RBFN. However, our MAPE is considerably lower than Pillo et. al [25]. A reason for that might be that this study used data from a food store department instead of a single product. Moreover the models in this study was trained on a larger data set which also might be a contributing factor to the generally lower error measurements. The performance of the SVM was not unexpected since Levis et al. [20] used a SVM to predict the monthly sales of a single product and showed a similar MAPE compared to this study. Our result is also in line with Krause-Traudes et al. [18] who concluded that SVMs can be used to predict aggregated sales in food a store. We did not expect the poor performance of the MLP, since multiple studies have shown its usage in sales prediction [10, 30, 29, 17]. The MLP shows surprisingly high standard deviations in both MAPE and RMSE compared to the other models. However our result could be a consequence of the parameter settings used since there are multiple ways to choose initial weights and the amount of neurons in the hidden layer.

**6 Conclusion**

Sales forecasting is very crucial for every company, especially for big ones. This process is very complex because there are a lot of factors that should be taken into account. In order to implement reachable goals and successfully achieve them, companies are keen on to predict next periods sales. Compared to other methods, ANNs are organic, and this method builds a learning algorithm to predict results better. In this study sales revenue forecasts are very close to actual sales revenues for each firm. Other factors that could affect sales revenue can be also put into the mix at further researchs. Also in this study only grocery retailing industry is analysed. In order to get a big picture, other industry‟s can be analysed at Turkey. And also, to compare the efficiency of the ANN for sales revenue,

**Reference**

[1] Kim Brynjolfsson Hitt. “Strength in Numbers: How Does DataDriven Decisionmaking Affect Firm Performance”. In: (2011). URL: <http://ebusiness.mit.edu/research/papers>

[2] Orinna Cortes and Vladimir Vapnik. “Support-vector networks”. In: Machine Learning 20(3) (1995), pp. 273–297.

[3] Nari Sivanandam Arunraj and Diane Ahrens. “A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting”. In: Internation Journal Production Economics 170 (2015), pp. 321–335.

[4] Philip Doganis et al. “Time series sales forecasting for short shelflife food products based on artificial neural networks and evolutionary computing”. In: Journal of Food Engineering 75 (2006), pp. 196–20.

[5] Maike Krause-Traudes et al. Spatial data mining for retail sales forecasting. Tech. rep. Fraunhofer-Institut Intelligente Analyse- und Informationssysteme (IAIS), 2008.

L. Breiman. Random forests. Machine Learning, 45:5–32, 2001.

L. Breiman. Consistency For a Simple Model of Random Forests. Technical Report 670, UC Berkeley, 2004. URL http://www.stat.berkeley.edu/˜breiman.

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. Classification and Regression Trees. Chapman & Hall, New York, 1984.

V. Svetnik, A. Liaw, C. Tong, J. Culberson, R. Sheridan, and B. Feuston. Random forest: A classification and regression tool for compound classification and QSAR modeling. Journal of Chemical Information and Computer Sciences, 43:1947–1958, 2003.

R. Diaz-Uriarte and S.A. de Andres. Gene selection and classification of microarray data using ´ random forest. BMC Bioinformatics, 7:1471–2105, 2006.

Y. Freund and R. Shapire. Experiments with a new boosting algorithm. In L. Saitta, editor, Machine Learning: Proceedings of the 13th International Conference, pages 148–156, San Francisco, 1996. Morgan Kaufmann

Z.-H. Zhou and M. Li. Ensemble Methods.(2012). Foundations and Algorithms, -13: 978-1-4398-3005 -5.

Jerome H. Friedman . (1999). Greedy Function Approximation: A Gradient Boosting Machine, IMS 1999 Reitz Lecture.